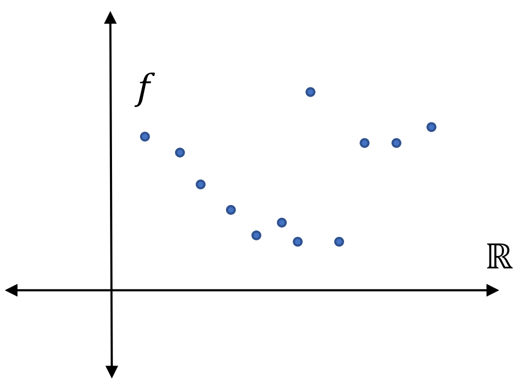
Introduction to Optimization Theory

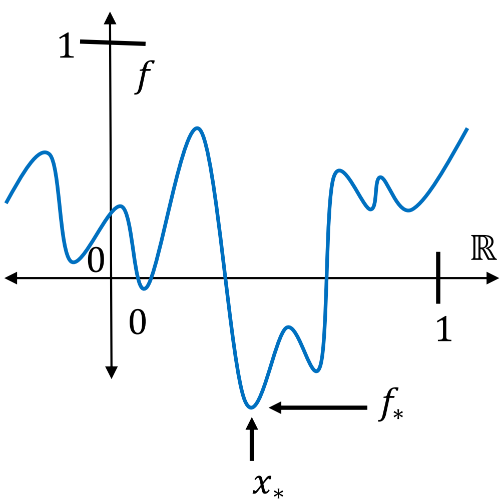
Lecture #19 - 11/19/20

MS&E 213 / CS 2690



Aaron Sidford

sidford@stanford.edu



Plan for Today

Recap

- Cutting plane methods

IPMs

- Strutured optimization
- Newton’s method
- Improving cutting plane methods
- Interior point methods

Have a great break!

Recap

Problem

$$\min_{x \in \mathbb{R}^n} f(x)$$

Regularity	Oracle	Goal	Algorithm	Iterations
$n = 1$, $f(x) \in [0,1]$, $x_* \in [0,1]$	value	½-optimal	anything	∞
$n = 1$, $x_* \in [0,1]$, L-Lipschitz	value	ϵ-optimal	ϵ-net	$\Theta(L/\epsilon)$
$x_* \in [0,1]$, L-Lipschitz in $\|\cdot\|_\infty$	value	ϵ-optimal	ϵ-net	$\left(\Theta(L/\epsilon)\right)^n$
L-smooth and bounded	value, gradient	ϵ-optimal	ϵ-net	exponential
L-smooth	gradient	ϵ-critical	gradient descent	$O(L(f(x_0) - f_*)\epsilon^{-2})$
L-smooth μ-strongly convex	gradient	ϵ-optimal	gradient descent	$O\big((L/\mu)\log([f(x_0) - f_*]/\epsilon)\big)$
L-smooth convex	gradient	ϵ-optimal	gradient descent	$O(L\|x_0 - x_*\|_2^2/\epsilon)$
L-smooth μ-strongly convex	gradient	ϵ-optimal	gradient descent	$O\big(\sqrt{L/\mu}\log([f(x_0) - f_*]/\epsilon)\big)$
L-smooth μ-strongly convex	gradient	ϵ-optimal	gradient descent	$O\left(\sqrt{L\|x_0 - x_*\|_2^2/\epsilon}\right)$
L-Lipschitz, convex	subgradient	ϵ-optimal	Mirror descent, FTRL	$O(L^2\|x_0 - x_*\|_2^2/\epsilon^2)$

Convex Function Oracle

Approach

- Reduce to feasibility problem
- Solve feasibility problem

query $x \in \mathbb{R}^n$ → **separation oracle** → output $g \in \mathbb{R}^n$ such that $\text{level}_{\leq}\big(f, f(x)\big) \subseteq H_{\geq}(g, g^\top x)$ With $g = 0$ only if $f_* = f(x)$

Cutting Plane Methods

query $x \in \mathbb{R}^n$ → **subgradient oracle** ✓ → output $g \in \partial f(x)$

Subgradient descent

Subgradient*: g is subgradient of f at x (i.e. $g \in \partial f(x)$) if and only if $f(y) \geq f(x) + g^\top (y - x)$ for all $y \in \mathbb{R}^n$*

Approach

- Reduce to online linear optimization
- Solve online linear optimization

Analogous role to online linear optimization problem

Box of radius R*:* $\{x \in \mathbb{R}^n \mid \|x - c\|_\infty \leq R\}$

(R, r, n)-Feasibility Problem

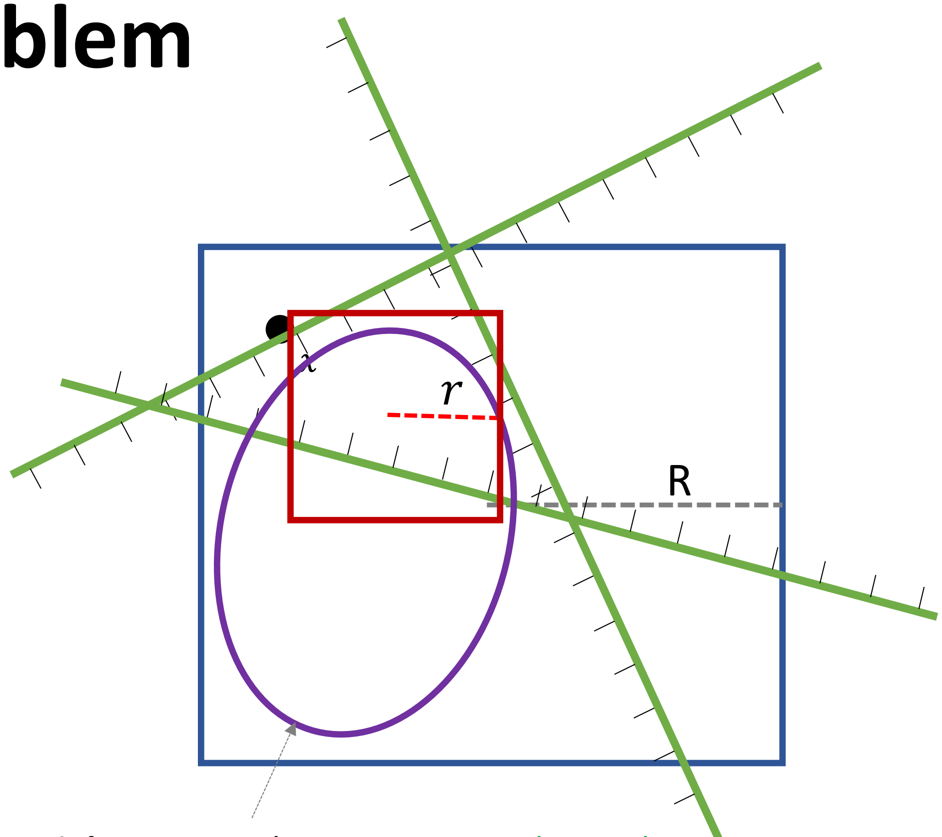
Given:

- n-dimensional box B of radius R:
- **Separation oracle**: when evaluated at point $x \in B$ in time T either outputs "success" or a halfspace with x on boundary.

Goal:

- Get "success" or
- Prove that the intersection of halfspaces and initial box does not contain a box of radius r

Note: Want $O(poly(n, T, \log(R/r))$



***Special Case**: we obtain separating hyperplanes for some convex set the is contain in a Box of radius R and contains a ball of radius r.*

Notation: $\kappa \overset{\text{def}}{=} nR/r$

Cutting Plane Methods

- Maintain a convex set
- Query separation oracle at "center" of convex set
- Update convex set
- Repeat until "size" is sufficiently small

Method	Set	Center	Size Tracked	Time ($\kappa = nR/\epsilon$)
Ellipsoid [YN76,S77,K80]	Ellipse	Ellipse	Volume	$O(n^2 T \log\kappa + n^4 \log\kappa)$
Center of Gravity [L65,BV02]	All half-spaces	Gravity	Volume	$O\left(nT\log\kappa + n^5 \log^{O(1)}\kappa\right)$
Inscribed Ellipse [KTE88,NN92]	All half-spaces	John Ellipse	Volume	$O(nT\log\kappa + n^{\omega+1.5}\log^{O(1)}\kappa)$
Volumetric Center [V89]	Some half-spaces	Volumetric	Volume, width	$O(nT\log\kappa + n^{\omega+1}\log\kappa)$
Analytic Center [AV95]	Some half-spaces	Analytic	Volume, width	$O(nT\log^{O(1)}\kappa + n^{O(1)}\log^{O(1)}\kappa)$
[LSW15]	**Some half-spaces**	**Hybrid**	**Volume, width**	$\boldsymbol{O(nT\log\kappa + n^3\log^{O(1)}\kappa)}$
[JLSW20]	**Some half-spaces**	**Volumetric**	**Volume, width**	$\boldsymbol{O\left(nT\log\kappa + n^3\log\kappa\right)}$

- Implication: L-smooth, μ-strongly convex ϵ-opt given ϵ_0-initial error with $O\left(n\log\left(\frac{nL\epsilon_0}{\mu\epsilon}\right)\right)$ gradient evaluations
- Implication: faster matroid intersection, semidefinite programming, submodular optimization

$\omega < 2.373$ [W14]

Plan for Today

Recap

- Cutting plane methods

IPMs

- Strutured optimization
- Newton's method
- Improving cutting plane methods
- Interior point methods

Have a great break!

Structured Convex Programming

Motivation

- Goal: $\min_{x \in \mathbb{R}^n} f(x)$

Want

- Linearly convergent algorithm
  - ϵ-optimal point in $O(\alpha \log(\epsilon_0/\epsilon))$ for problem dependent (ideally small) α

Before

- GD: $O((L/\mu) \log(\epsilon_0/\epsilon))$
- AGD: $O((\sqrt{L/\mu}) \log(\epsilon_0/\epsilon))$
- Cutting Plane: ~$O(n \log(\epsilon_0/\epsilon))$

Question

- Can we improve if have more structure?
- Can we reduce to more difficult subproblem (than separation) and have less iterations?

General Problem

Linear Optimization / Convex Programming

- $\min_{x \in S \subseteq \mathbb{R}^n} c^\top x$ for convex S

Why?

- $\min_{x \in \mathbb{R}^n} f(x) \Leftrightarrow \min_{f(x) \le t} t \Leftrightarrow \min_{(x,t) \in S} t$ where $S = \text{epi}(f)$

Hope

- Leverage structure of S
- Get better rates
- **Spoiler**: there is theory supporting $O(\sqrt{n} \log(\epsilon_0/\epsilon))$

Linear Programming

Input

- $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$

Goal

- $\min_{x \in P} c^\top x$ for $P \stackrel{\text{def}}{=} \{x : Ax \geq b\}$
- $= \min_{x \in \mathbb{R}^n} c^\top x + \psi_P(x)$ for $\psi_P(x) \stackrel{\text{def}}{=} \begin{cases} 0 & Ax \geq b \\ \infty & \text{otherwise} \end{cases}$

The Picture

(Closed) Half-space

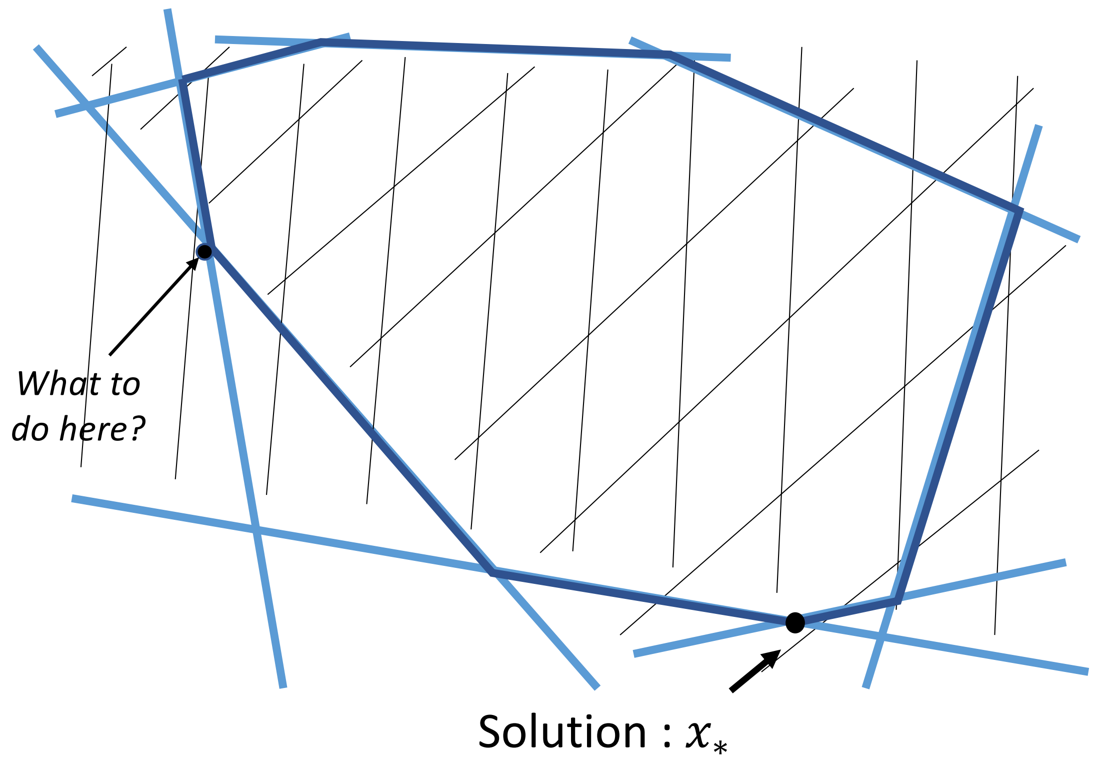
$$\mathrm{half}(a_i, b_i) \stackrel{\text{def}}{=} H_{\geq}(a_i, b_i) \stackrel{\text{def}}{=} \{x \in \mathbb{R}^n : a_i^\top x \geq b_i\}$$

$$\min_{x \in \mathbb{R}^n :\, Ax \geq b} c^\top x$$

Polytope $Ax \geq b$

$$\begin{pmatrix} - & a_1 & - \\ - & a_2 & - \\ & \vdots & \\ - & a_k & - \\ & \vdots & \\ - & a_m & - \end{pmatrix} x \geq \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_k \\ \vdots \\ b_m \end{pmatrix}$$

$c^\top x$



***Recall**: as $m \to \infty$ problem converges to arbitrary convex programming*

Feasibility is as Difficult as Optimization

Optimization

$$\min_{x \in \mathbb{R}^n:\, Ax \geq b} c^\top x$$

Feasibility

Find $x \in \mathbb{R}^n$ with $Ax \geq b$

Why?

$$A'x \geq b' = \begin{Bmatrix} c^\top x \leq t \\ Ax \geq b \end{Bmatrix}$$

and binary search on t

Note

More difficult than solving $Ax = b$

$$\begin{pmatrix} A \\ -A \end{pmatrix} x \geq \begin{pmatrix} b \\ -b \end{pmatrix} \Leftrightarrow \begin{matrix} Ax \geq b \\ Ax \leq b \end{matrix} \Leftrightarrow Ax = b$$

Algorithms today will assume one

Assuming a Feasible Point

$$\min_{x \in \mathbb{R}^n :\, Ax \geq b} c^\top x$$

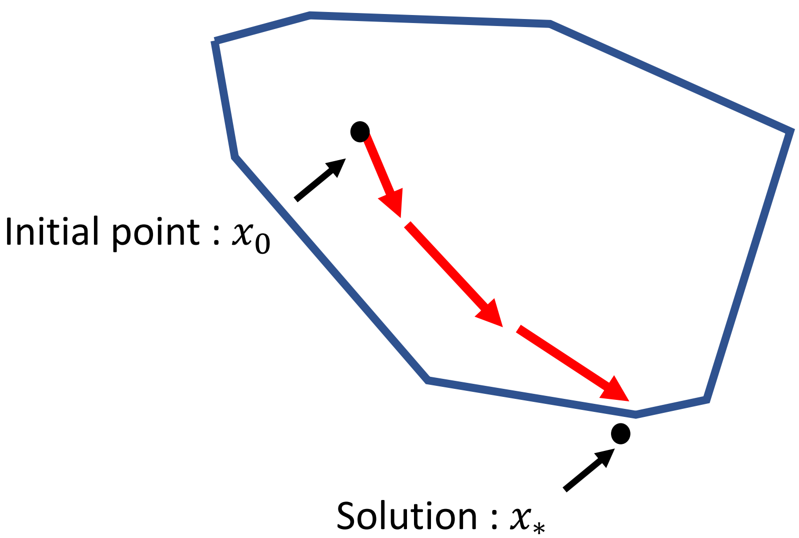
How?

(here is one simple transformations)

$$\min_{Ax + \vec{1}\alpha \geq b \text{ and } \alpha \geq 0} c^\top x + M \cdot \alpha$$

New Problem

<u>Goal</u> given x_0 such that $Ax_0 > b$ solve $\min_{x \in \mathbb{R}^n:\ Ax \geq b} c^\top x$



<u>How?</u>

- iterative algorithm
- Improve quality each step
- **Idea**: more powerful oracle

<u>Problem</u>: improving is difficult at boundary
<u>Solution</u>: use barrier to penalize approaching boundary

<u>Problem</u>: GD converges fast when level sets are ~ ℓ_2 balls
<u>Solution</u>: work in local norm – Newton's method

Core concept explicit / implicit in many IPMs

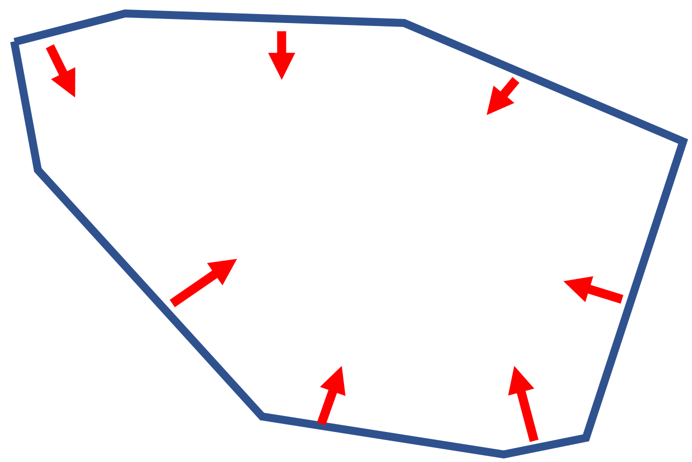
Barrier Function

$$\min_{x\,:\,Ax \geq b} c^{\top} x$$

Barrier function

A “nice” function p from interior to $\mathbb{R}$ s.t.

$$\lim_{x \to boundary} p(x) \to \infty$$



Idea

- Trade off minimizing cost and minimizing barrier

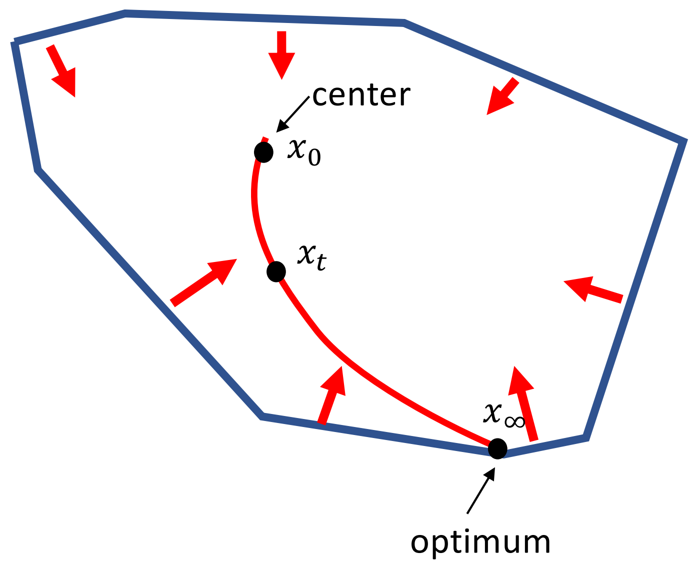
Are many ways of using barrier, this is just one

Path Following Methods

One example in larger active area of research on "higher order" optimization methods.

Approach

- Approximately follow the central path
- Start near the path
- Take step to converge to next path point
- Repeat



How take step?

- Gradient descent / steepest descent in "local norm"
- Want $z^\top \nabla f(x) z \approx ||z||^2$ so pick $||z|| = ||z||^2_{\nabla f(x)}$
- (Damped Newton Step) $x := x - \eta\left(\nabla^2 f_t(x)\right)^{-1} \nabla f_t(x)$

$$\min_{x\,:\,Ax \geq b} c^\top x$$

Barrier function

A "nice" function p from interior to $\mathbb{R}$ s.t.

$$\lim_{x \to boundary} p(x) \to \infty$$

Penalized Objective

$$f_t(x) = t \cdot c^\top x + p(x)$$

Central Path

For path parameter $t > 0$ the minimizers $x_t = \mathrm{argmin}_x f_t(x)$ form the *central path* a continuous curve from *center* (x_0) to solution (x_∞).

Discretization* of the **central path**

Are many ways of using barrier, this is just one

Path Following Methods

$$\min_{x\,:\,Ax\geq b} c^\top x$$

Barrier function

A "nice" function p from interior to $\mathbb{R}$ s.t.

$$\lim_{x\to boundary} p(x) \to \infty$$

Penalized Objective

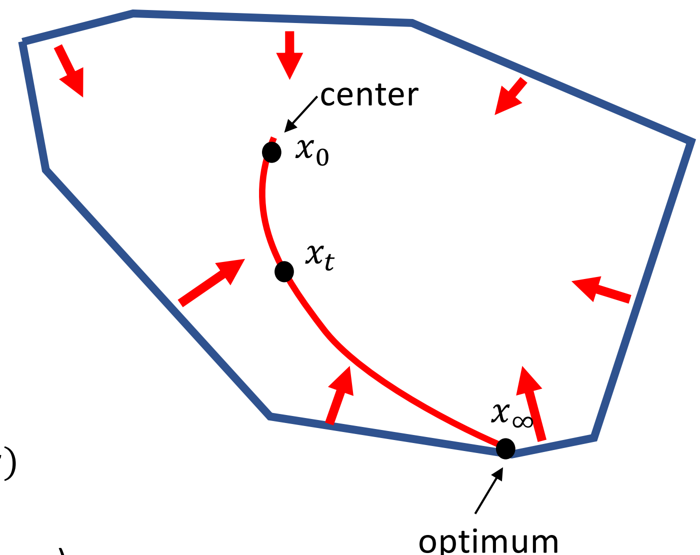
$$f_t(x) = t \cdot c^\top x + p(x)$$

Central Path

For path parameter $t > 0$ the minimizers $x_t = argmin_x f_t(x)$ form the *central path* a continuous curve from *center* (x_0) to solution (x_∞).

Algorithm

- **Initialize**: $t > 0$ and $x \approx x_t$
- **Iterate**: repeat until $c^\top x$ small
- **Move path parameter**
  - $t \coloneqq (1+c)t$
- **Center (i.e. Newton Steps)**
  - Until $x \approx x_t$
  - $x \coloneqq x - \eta\left(\nabla^2 f_t(x)\right)^{-1} \nabla f_t(x)$



How get to initial point? (one technique)

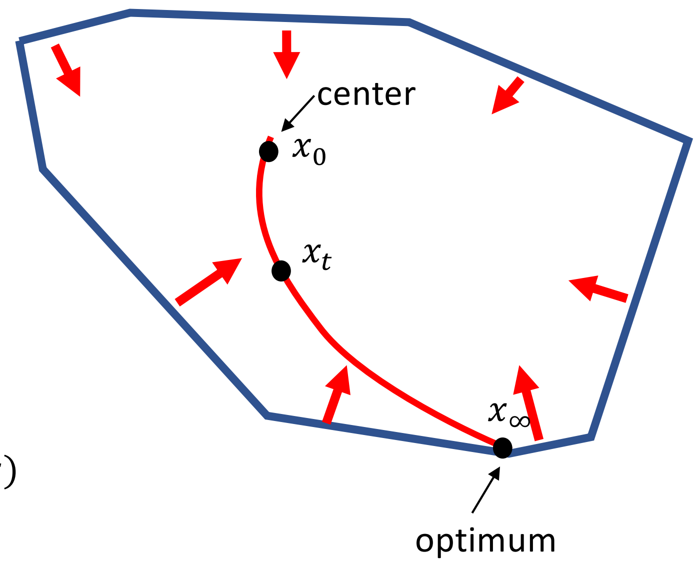
- Initial x_0 is central path for some c
- Let $t \coloneqq (1-c)t$
- When t small, switch cost

Discretization* of the **central path**

Analysis

Algorithm

- **Initialize**: $t > 0$ and $x \approx x_t$
- **Iterate**: repeat until $c^\top x$ small
- **Move path parameter**
  - $t \coloneqq (1 + c)t$
- **Center (i.e. Newton Steps)**
  - Until $x \approx x_t$
  - $x \coloneqq x - \eta\left(\nabla^2 f_t(x)\right)^{-1} \nabla f_t(x)$



Self-conccordance

- If p is a ν-*self concordant barrier* then can pick $c \approx \frac{1}{O(\sqrt{\nu})}$ and $O(\sqrt{\nu}\log(\epsilon/\epsilon_0))$ iterations suffice
- Self concordance measures tradeoff of Hessian stability and size
- **Theorem**: every convex set has an $O(d)$ self-concordant barrier (and can be better sometimes)
- **Upshot**: if structure, barrier then $\left(\nabla^2 f_t(x)\right)^{-1} \nabla f_t(x)$ queries suffice

$$\min_{x\,:\,Ax \geq b} c^\top x$$

Barrier function

A "nice" function p from interior to $\mathbb{R}$ s.t.

$$\lim_{x \to boundary} p(x) \to \infty$$

Penalized Objective

$$f_t(x) = t \cdot c^\top x + p(x)$$

Central Path

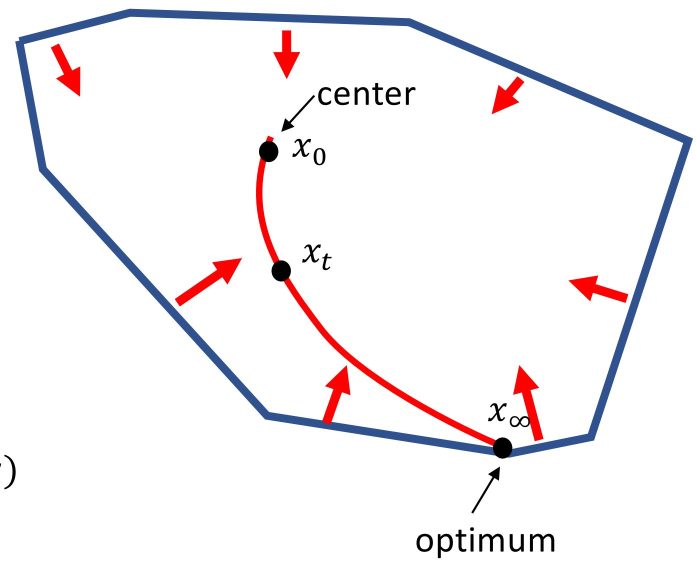
For path parameter $t > 0$ the minimizers $x_t = argmin_x f_t(x)$ form the *central path* a continuous curve from *center* (x_0) to solution (x_∞).

Discretization* of the **central path**

Example

Algorithm

- **Initialize**: $t > 0$ and $x \approx x_t$
- **Iterate**: repeat until $c^\top x$ small
- **Move path parameter**
  - $t := (1 + c)t$
- **Center (i.e. Newton Steps)**
  - Until $x \approx x_t$
  - $x := x - \eta\left(\nabla^2 f_t(x)\right)^{-1} \nabla f_t(x)$



$$\min_{x\,:\,Ax \geq b} c^\top x$$

Barrier function

A "nice" function p from interior to $\mathbb{R}$ s.t.

$$\lim_{x \to boundary} p(x) \to \infty$$

Penalized Objective

$$f_t(x) = t \cdot c^\top x + p(x)$$

Central Path

For path parameter $t > 0$ the minimizers $x_t = argmin_x f_t(x)$ form the *central path* a continuous curve from *center* (x_0) to solution (x_∞).

Discretization* of the **central path**

Logarithmic Barrier

- $p(x) = -\sum_{i \in [n]} \ln\left(\frac{1}{a_i^\top x - b_i}\right) = \sum_{i \in [n]} -\ln(s(x)_i)$ where $s(x) = Ax - b$ is a $O(m)$ self-concordant barrier
- $\nabla p(x) = -A^\top S_x \vec{1}$ and $\nabla^2 p(x) = A^\top S_x^{-2} A$ where $S_x = \text{diag}(s(x))$
- Upshot: ~$O(\sqrt{m}\log(\epsilon/\epsilon_0))$ linear system solves suffice

What Else?

Active Area of Research

- Improve iteration count
- Decrease iteration cost
- Multiple recent results
- Improvements for special important cases (e.g. maximum flow, geometric median, etc.)

More Optimization Theory

- Higher-order methods
- Structured problems
- Minimax problems
- Stochastic
- Nonconvex
- Practice
- And more…

Plan for Today

Recap

- Cutting plane methods

IPMs

- Strutured optimization
- Newton’s method
- Improving cutting plane methods
- Interior point methods

Have a great break!